2021 IEEE International Conference on Smart Internet of Things (SmartIoT) | 978-1-6654-4511-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/SmartIoT52359.2021.00025

Monocular Pseudo-LiDAR 3D Object Detection Method Based on Confidence and Feature Optimization

Jianlong Zhang
School of Electronic Engineering
Xidian University
Xi'an, China
jlzhang@mail.xidian.edu.cn

Guangzu Fang
School of Electronic Engineering
Xidian University
Xi'an, China
gzfang@stu.xidian.edu.cn

Bin Wang
School of Electronic Engineering
Xidian University
Xi'an, China
bwang@xidian.edu.cn

Chen Chen
State Key Laboratory of Integrated Services Networks
Xidian University
Xi'an, China
cc2000@mail.xidian.edu.cn

Xinyu Guo
School of Electronic Engineering
Xidian University
Xi'an, China
gxy8696@126.com

Yang Zhou
The Ministry of water resources of China
Beijing, China
zhy@mwr.gov.cn

Ji Li
The Goldenwater Information Technology Co. Ltd.
Beijing, China
liji@goldenwater.com.cn

***Abstract*—3D object detection based on the monocular Pseudo-LiDAR is cost-effective compared with multi-sensor solutions in autonomous driving, and made great progress in recent. However, these methods still suffer from the following disadvantages, i.e., (1) the depth estimation error of the background points in the Pseudo-LiDAR is larger than that of the object points, which leads to the degradation of the object detection performance. (2) the existing point cloud detection network is difficult to effectively capture the local features and global correlation of Pseudo-LiDAR, which leads to the weak feature representation ability of the point cloud. To this end, we propose a 3D object detection method of Pseudo-LiDAR based on Confidence and Feature optimization. In our method, we first propose a hierarchical structure of the point cloud confidence optimization method, which carries out resampling to filter out the background points according to the confidence distribution, so as to improve the significance of the Pseudo-LiDAR object point cloud. Then, we design a hierarchical feature extraction module, which uses DGCNN to obtain the local features of the keypoints and the point transformer structure to capture the global relevance of the keypoints, so as to improve the feature representation ability of the point cloud. On the popular KITTI benchmark, our approach achieves better performance than the mainstream method.**

***Keywords*—Monocular 3D object detection, Pseudo-LiDAR, Confidences Optimization, Hierarchical Feature Extraction**

I. Introduction

In recent years, 3D object detection is the key technology in many applications such as robotic and autonomous driving applications and is a more realistic way to describe the object. At present, the associated methods to obtain 3D scene information could be categorized into LiDAR-based methods, image-based methods and multiple sensors-based methods in autonomous driving solutions. The quality of the point cloud generated by the LiDAR sensor depends heavily on the characteristics of the laser scanning, e.g., the number and rotation speed of the laser beam [1]. LiDAR sensors are expensive due to the limitation of material and technology, leading to a high cost of multi-sensor solution. So many studies began to replace them with cheap cameras [1].

At present, monocular camera [2], stereo camera and depth camera [3] are the main sensors in image-based 3D detection. The monocular image is limited by the lack of depth information, and its performance is poor compared with the other two schemes. However, the advantages like mature technology, low cost, and suit for industrialized production still make it receive extensive attention. 3D object detection based on monocular images is a huge challenge, so some methods MonoGRNet [4] and M3D-RPN [5] try to extend 2D object detection from images to 3D object detection space. Although these methods have achieved certain results, their scalability is far from satisfying.

Recently, scholars have proposed the 3D object detection method based on Pseudo-LiDAR. The method uses the mature depth estimation network to generate the monocular depth map from the monocular image and transforms it into the 3D point cloud to represent the spatial information in the scene, then took advantage of LiDAR-based methods to acquire 3D bounding boxes. Pseudo-lidar [6] utilized monocular depth estimation methods to generate Pseudo-LiDAR point cloud and then employed Frustum PointNet 3D detection framework to predict 3D bounding boxes. AM3D used estimated depth to generate point cloud data, and utilized an attention mechanism to guide the message passing, finally made a significant improvement on 3D object detection performance [7]. But [6,7] relies heavily on the Pseudo-LiDAR generated by monocular depth estimation, and the quality of the generated point cloud directly affects the final detection performance. Due to the influence of depth estimation error, the detection performance of these methods is poor, and there is a large space for improvement.

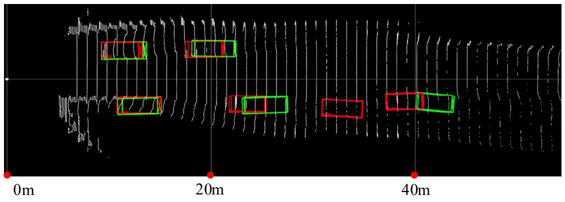

Fig. 1. Sample KITTI [8] 3D detection results.

Fig. 1 depicts the typical result of a bird's-eye view of Pseudo-LiDAR on KITTI3D [8] validation for "Car". Ground truth boxes are in green and predicted boxes are in red. It shows obviously that the detection performance of the object decreases sequentially as the object distance increases.

978-1-6654-4511-5/21/$31.00 ©2021 IEEE
DOI 10.1109/SmartIoT52359.2021.00025

This is caused by the depth estimation error of the long distant background point cloud. Moreover, the LiDAR-based detection network still has great potential over-extraction methods of point cloud feature, for example introducing Transformer [9] which enjoys excellent global correlation performance.

A monocular 3D object detection method based on the confidence and feature optimization of Pseudo-LiDAR is proposed. The main contributions of this paper can be summarized as follows: (1) We introduce PV-RCNN [10] as the detection network for monocular 3D detection, which improves the benchmark of monocular 3D detection. (2) We propose a hierarchical structure of the point cloud confidence optimization method to improve the significance of the object point cloud according to the confidence distribution and reduce the influence of depth estimation error. (3) We design a hierarchical feature extraction module, which effectively takes into account both the local and global features, and improves the feature representation ability of the point cloud.

The rest of this paper is organized as follows. In Section II, We introduce the principles and implementation of the proposed method. Section III provides experimental evaluations and analysis. Finally, we conclude our work in Section IV.

II. Approach

The proposed monocular 3D detection framework, as shown as Fig. 2, mainly includes two parts, i.e., (1) Pseudo-LiDAR generation and optimization, and (2) Pseudo-LiDAR object detection. In the generation and optimization module, the 3D visual point cloud is generated first based on a monocular depth estimation network, then the visual point cloud is mapped from camera coordinate system to LiDAR coordinate system, and finally, the confidence optimization method is adopted to improve the object's saliency. In Pseudo-LiDAR object detection, we use PV-RCNN as our backbone net and introduce DGCNN [11] and Point Transformer [12] to construct a hierarchical feature extraction module to obtain the local and global features of the point cloud, respectively.

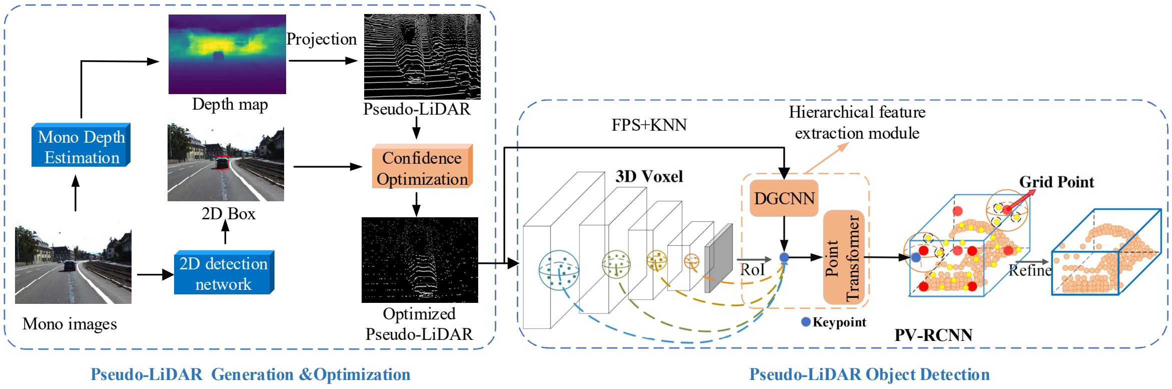

Fig. 2. The overall architecture of our proposed monocular 3D detection framework. (*FPS: farthest point sampling, KNN: K-nearest neighbor.*)

A. Pseudo-LiDAR generation

We firstly use DORN [13] as the depth estimation network of monocular images, and this network uses ordinal regression model to improve the accuracy of the long-distance depth estimation. And then the depth map is mapped to the 3D visual point cloud via the projection transformation. Finally, the visual point cloud is converted from the camera coordinate to the LiDAR coordinate by the 3D coordinate transformation to obtain the Pseudo-LiDAR, as illustrated in Fig. 3.

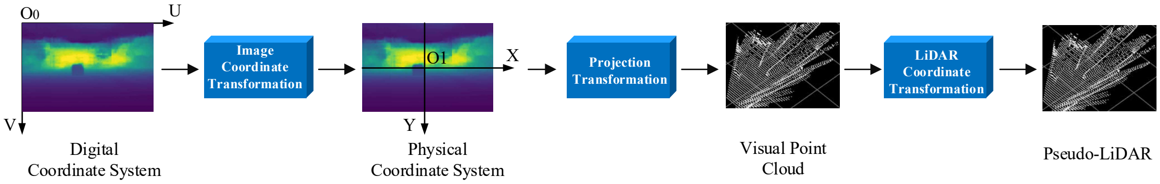
O0
U
V
Image Coordinate Transformation
O1
X
Y
Projection Transformation
LiDAR Coordinate Transformation
Digital Coordinate System
Physical Coordinate System
Visual Point Cloud
Pseudo-LiDAR

Fig. 3. Detailed structure design for Pseudo-LiDAR generation.

B. Confidence optimization

Monocular depth estimation algorithm has the following two defects, i.e., 1) the depth value on the edge of the object vehicle is affected by the background, which results in the increase of depth estimation error in the transition region between the object and the background, 2) the depth estimation error of background points is much larger than that of object points, and increases nonlinearly with the distance. Therefore, we propose a hierarchical method for point cloud confidence optimization, which randomly resamples the Pseudo-LiDAR according to the local and global confidence to filter out the background points with large depth estimation error, so as to improve the significance of object points.

The confidence optimization method, as illustrated in Fig. 4, mainly consists of the following parts, i.e., 1) the 2D detection module for localizing the position of object vehicle in 2D image and distinguishing the object and the background; 2) the local confidence module (LCM) for reducing the confidence of the points in transition part between the object and the background; 3) the global confidence module (GCM) for reducing the confidence of the long-distance points; 4) the confidence resampling module for filtering out the background points with large depth estimation error.

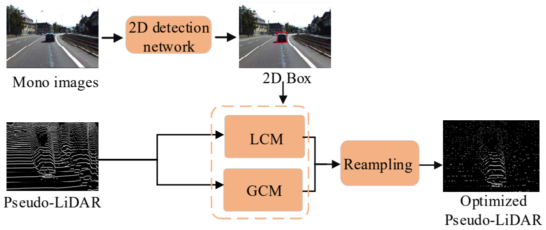

Fig. 4. Confidence optimization method.

1) 2D detection network: Cascade R-CNN [14], a detection model, is taken as our 2D detection network. Compared with the traditional R-CNN network, it detects objects multiple times in a cascading manner and trains a high-performance 2D detection network without changing the number of samples. It is worth noticing that the over-fitting problem herein caused by the cascade network is reduced by re-detection.

2) Local confidence module (LCM): Assuming that the local confidence in the detection bounding box to obey 2D Gaussian distribution, we take the center of the 2D detection bounding box as the anchor point and compute the local confidence according to the distance between the point cloud and the anchor point. This method reduces the confidence of the vehicle edge point cloud, and weakens the influence of the depth estimation error on the detection.

The spatial coordinate of a point in the Pseudo-LiDAR point cloud is denoted by $p(x,y,z)$, and its projection point on the 2D image coordinate system is defined as $a(u,v)$. If projection point a is located inside the 2D detection bounding box b, the weight of the Gaussian distribution of the projection point a relative to b is $\alpha_{(a,b)}$ defined as

$$\alpha_{(a,b)} = \begin{cases} s(u,v) & y(a,b)=1 \\ 0 & y(a,b)=0 \end{cases}, \tag{1}$$

where $y(a,b)=1$ means that a is located inside the detection bounding box b. The calculation formula of $s(u,v)$ is

$$s(u,v) = \frac{1}{2\pi\sigma^2} e^{-\frac{(u-u_c)^2+[(v-v_c)*\frac{w}{h}]^2}{2\sigma^2}}, \tag{2}$$

where w and h are the pixel width and height of the detection bounding box b, respectively, decay rate parameter σ is set to $w/5$, and (u_c,v_c) is the center coordinate of the detection bounding box b.

When the object vehicle is occluded, some 2D detection bounding boxes overlap each other, and a single point cloud p may be transformed into several bounding boxes due to multiple detection. At this time, the maximum confidence weight is used to calculate the weight of a single point cloud p. Assume that $\mathbf{B}=\{b_1,b_2,\ldots,b_m\}$ is the set containing all the detection bounding boxes of projection point a in the image scene, the weight $\alpha_{(a,\mathbf{B})}$ of projection point a is

$$\alpha_{(a,\mathbf{B})} = \max(\alpha_{(a,b)}), b \in \mathbf{B}. \tag{3}$$

Then the local confidence $S_{Loc}(p)$ of the Pseudo-LiDAR p is

$$S_{Loc}(p) = \max(f_{norm}(\alpha_{(a,\mathbf{B})}), \xi_\alpha), \tag{4}$$

where f_{norm} is the scene weight normalization function of the Pseudo-LiDAR point cloud set, and ξ_α is the local background threshold so that the background points outside the detection bounding boxes will not be completely filtered out.

3) Global confidence module (GCM): Due to the lack of depth prior information in monocular images, the depth estimation error increases nonlinearly as the distance of the scene increases. Considering that, we design a confidence decay rate R_γ, which makes the global confidence distribution decrease with the increase of distance and weakens the influence of long-distance depth estimation error on detection.

Since the depth distribution of the Pseudo-LiDAR in different scenes is quite different, the Pseudo-LiDAR of the current scene is defined as $Q(p)$. We first introduce the global balance parameter λ_β, and then calculate the confidence decay rate R_γ according to the point cloud depth distribution as

$$R_\gamma = \frac{1}{\lambda_\beta f_E(Q) + f_D(Q)}, \tag{5}$$

where $f_E(Q)$ and $f_D(Q)$ are the depth mean and the depth variance of the Pseudo-LiDAR, respectively.

$$f_E(Q) = \frac{\sum\limits_{p\in Q} d_p}{|Q|}, \quad f_D(Q) = \sqrt{\frac{\sum\limits_{p\in Q}[d_p - f_E(Q)]}{|Q|}}, \tag{6}$$

where d_p is the depth value of point cloud p.

Then the global confidence calculation formula of the Pseudo-LiDAR point cloud is

$$S_{Global}(p) = \max(1 - R_\gamma d_p, \xi_\beta), \tag{7}$$

where ξ_β is the global background threshold.

Equation (7) indicates that $S_{Global}(p)$ decays with the increase of point cloud depth at a rate of R_γ. The background threshold ξ_β ensures that the point cloud at a long distance is not be completely filtered out.

The final Pseudo-LiDAR point cloud confidence $S(p)$ is generated by weighting $S_{Global}(p)$ to $S_{Loc}(p)$, i.e.,

$$S(p) = S_{Loc}(p) \cdot S_{Global}(p). \tag{8}$$

4) Confidence resampling: Each point in the raw Pseudo-LiDAR set Q_{raw} is resampled according to the corresponding confidence, and the resampled point cloud set is Q_{re} defined as

$$Q_{re} = \{p \mid S(p) > rand(0,1), p \in Q_{raw}\}. \tag{9}$$

The results before and after confidence resampling of the Pseudo-LiDAR are illustrated in Fig. 5 left and right, respectively. It can be seen that the background and object edge points become sparse, and the saliency of the object point cloud is also improved.

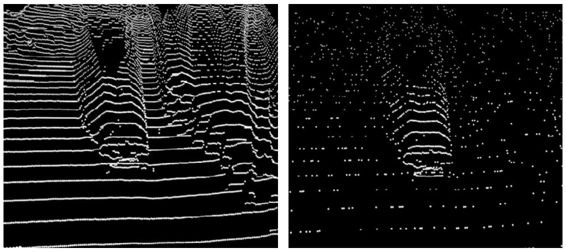

Fig. 5. Confidence resampling. (*Raw Pseudo-LiDAR*(*left*). *Resampled Pseudo-LiDAR*(*right*).)

C. Pseudo-LiDAR object detection network based on hierarchical feature extraction structure

We propose a Pseudo-LiDAR object detection network based on PV-RCNN [10]. PV-RCNN refines 3D box proposals by extracting the feature of keypoints. So the quality of keypoint features directly affects the final detection performance. Each keypoint is obtained by the farthest point sampling (FPS), but the sparse sampling process leads to feature loss. And the features of each keypoint are extracted by the PointNet-based set abstraction which fails to consider the local neighborhood features and global relevance of the point cloud. It can be seen from the above that the raw PV-RCNN network could not effectively extract the features of the Pseudo-LiDAR. So we propose a hierarchical feature extraction structure based on PV-RCNN object detection network and a feature extraction module illustrated Fig. 6 is mainly composed of DGCNN [11] and Point Transformer [12] structure, which obtain local and global features of keypoints, respectively.

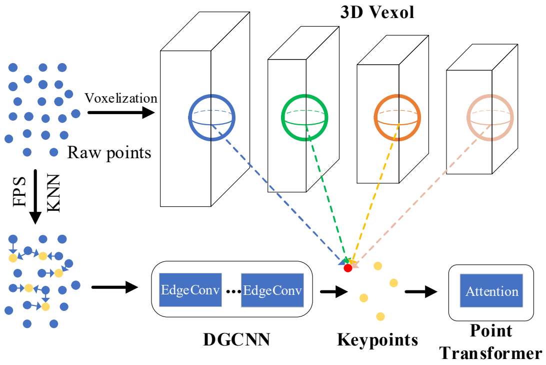

Fig. 6. Hierarchical feature extraction module.

1) Local feature extraction: Firstly, the local neighborhood structure of keypoints is constructed by K-nearest neighbor(KNN), and the features of the raw points are aggregated on the keypoints, so as to the sampling keypoints contain a lot of feature information of the raw point cloud. Secondly, EdgeConv in DGCNN network is used to extract local neighborhood features of the keypoints, and the local neighborhood features of keypoints are mapped to high-dimensional local feature space by cascading EdgeConv.

The definition of EdgeConv is as follows. $e_{ij}=h_\Theta(x_i,x_j\text{-}x_i)$ is the edge feature, where h_Θ is some parametric non-linear function parameterized by the set of learnable parameters Θ [11]. A is max pooling aggregation operation, then the operation of max-pooling of edge features is called EdgeConv, as shown in equation (10):

$$x_i^{'} = \underset{j:(i,j)\in\varepsilon}{\mathrm{A}} h_\Theta(x_i, x_j\text{-}x_i), \tag{10}$$

where $\{j : (i, j)\in\varepsilon\}$ as the patch around the central pixel x_i(see Fig. 7). The output of EdgeConv is calculated by aggregating all edge features, as illustrated in Fig. 7.

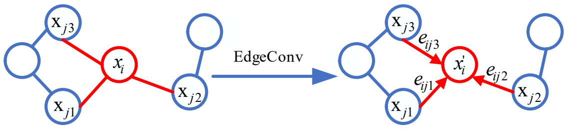

Fig. 7. Visualize the EdgeConv operation.

2) Global feature extraction: In this paper, the Point Transformer structure is introduced to improve the ability of PV-RCNN network to capture the global correlation characteristics of keypoints and improve the network's perception of the object geometry. Transformer [9] is an outstanding contribution in NLP in recent years. It mainly uses self-attention mechanism to capture the long-term correlation in sequences. Recent research has extended it to the field of point cloud segmentation by variant transformer [12]. Point Transformer uses vector attention, and its vector attention weight can adjust multiple feature channels. Because point clouds are essentially irregular sets embedded in the metric space, self-attention is naturally applicable to point clouds.

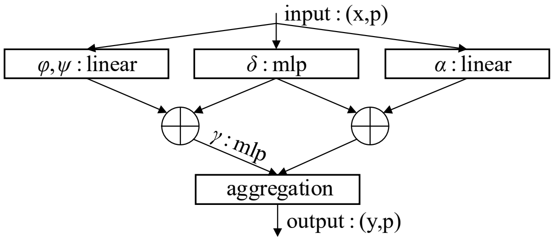

Fig. 8. Point Transformer layer.

The attention expression of Point Transformer is

$$y_i = \sum_{x_j\in\chi(i)} \rho\Big(\gamma\big(\varphi(\mathrm{x}_i)-\psi(\mathrm{x}_j)+\delta\big)\Big)\odot\big(\alpha(\mathrm{x}_j)+\delta\big), \tag{11}$$

where $\chi=\{x_i\}_i$ is the feature vector set of keypoints. y_i is the output feature of keypoints. ρ is a normalization function (e.g., a *softmax*). γ is a mapping function such as MLP. φ, ψ and α are the feature transformations of keypoints. δ is a position encoding function. The structure of the Point Transformer layer is shown in Fig. 8 [12].

Since point transformer consumes a lot of GPU memory, we chose FPS to down-sample the keypoints, and then the keypoints are output to the point transformer layer to obtain the long-distance interaction between the keypoints and the global features.

III. Experiments

A. Experimental details

1) Datasets: KITTI [8], the most popular dataset in 3D automatic driving, is used for training and testing, and has 7481 training samples and 7518 test samples. In order to test the performance of our network, the training samples are divided into the training set (3712 samples) and validation set (3769 samples), which are used to train and test the performance of our network model.

2) Parameter setting: In the calculation of the local confidence, the local background threshold ξ_α is set to 0.2. In the calculation of the global confidence, the global balance parameter λ_β is set to 1.5 and the global background threshold ξ_β is set to 0.2. The number of sampling points is 2048, and the number of nearest neighbors K is 16.

3) Training: We code our deep learning framework by Pytorch. We use Adam optimizer that is decayed according to the cosine annealing strategy. We trained the network on a GTX 2080Ti GPU.

4) Metric: We use the official evaluation tool provided by KITTI benchmark to calculate the 3D Average Precision under the Intersection of Union (IOU) threshold of 0.7 and use 11 interpolation points (AP_{11}) to calculate the AP value of "Car" categories. Note that the benchmark divides "Car" category into three cases: easy, moderate and hard according to the bounding box height and occlusion level.

B. Performance comparison

The 3D "Car" object detection results of our method and the four mainstream monocular 3D detection methods MonoGRNet, M3D-RPN, pseudo-LiDAR, and AM3D are shown in Table 1. The performance of our method is superior to the other monocular detection methods under the evaluation index of AP_{11} and IOU greater than 0.7. In the three cases(easy, moderate, hard), compared with the best AM3D, the AP_{11} value of our method is increased by **6.92%**, **5.76%**, and **5.39%**, respectively.

TABLE I. KITTI 3D "Car" Validation Set

Method	*Easy(%)*	*Moderate(%)*	*Hard(%)*
MonoGRNet [4]	13.88	10.19	7.62
M3D-RPN [5]	20.27	17.06	15.21
Pseudo-lidar [6]	28.20	18.50	16.40
AM3D [7]	32.23	21.09	17.26
Our method	**39.15**	**26.85**	**22.65**

C. Ablation experiment

We conduct the following ablation studies, i.e., (1) removing the hierarchical feature extraction module(HFEM) and the confidence optimization module(COM) from our method, (2) removing the hierarchical feature extraction module(HFEM) from our method. The results of ablation analysis are shown in Table 2. The best-performed model is the combination of using the COM and the HFEM, i.e., the proposed method. As shown in Table 2, both the COM and the HFEM can improve the performance of monocular 3D detection. The 3D detection results of the above three experiments are shown in Fig. 9, ground truth boxes in green and predicted boxes in red. In the network with the COM, the number of the false-positive detection bounding box is reduced, which indicates that can reduce the influence of depth estimation error and improve the detection performance. After adding the HFEM, the detection accuracy of the long-range object is improved, and the overall detection performance is also improved.

TABLE II. KITTI 3D "Car" Validation Set

Method	*Easy(%)*	*Moderate(%)*	*Hard(%)*
(1) Remove HFEM and COM	35.24	24.55	20.75
(2) Remove HFEM	36.96	25.84	21.13
(3)The full framework	**39.15**	**26.85**	**22.65**

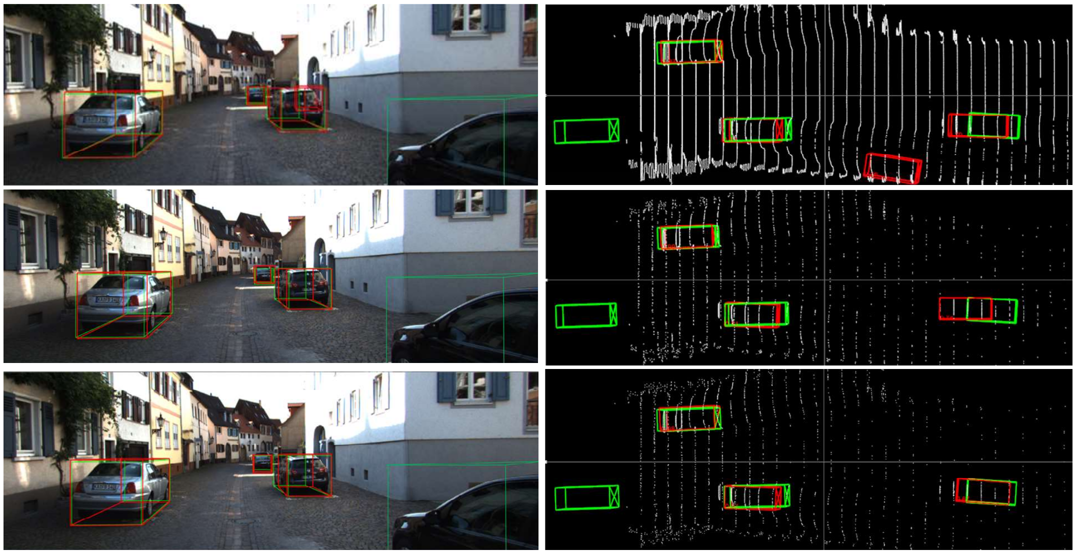

Fig. 9. Qualitative comparisons of 3D detection results. (*The 3D detection results of RGB image* (*Left*), *the 3D detection results of Pseudo-LiDAR* (*Right*). *Removing HFEM and COM* (*top*), *removing HFEM* (*middle*), *and the full framework* (*bottom*).)

IV. CONCLUSION

In this paper, we propose a monocular 3D detection method based on the confidence and feature optimized Pseudo-LiDAR. We use PV-RCNN as the backbone network for object detection, which improves the detection ability of the Pseudo-LiDAR object. The confidence optimization enhances the significance of the object point cloud, and the hierarchical feature extraction structure improves the feature representation ability of keypoints. The simulation experiment results show that the proposed method has obvious advantages in the three cases of the “Car” object on KITTI compared with the mainstream monocular 3D detection methods. Furthermore, the detection performance of our detection network could be better given a more accurate depth estimation map.

ACKNOWLEDGMENT

This work was supported by the Aeronautical Science Foundation of China (No.2018ZC81001), the National Natural Science Foundation of China (No.61971331), the National Key Research and Development Program of China (2018YFE0126000), the National Natural Science Foundation of China (62072360, 61902292, 62001357, 62072359, 62072355), the Key Research and Development Plan of Shaanxi province (2021ZDLGY02-09, 2019ZDLGY13-07, 2019ZDLGY13-04, 2020JQ-844), the Key Laboratory of Embedded System and Service Computing (Tong ji University) (ESSCKF2019-05), Ministry of Education, the Xi'an Science and Technology Plan (20RGZN0005) and the Xi'an Key Laboratory of Mobile Edge Computing and Security (201805052-ZD3CG36).

REFERENCES

[1] Vianney, J. M. Uwabeza, S. Aich, and B. Liu, “RefinedMPL: Refined Monocular PseudoLiDAR for 3D Object Detection in Autonomous Driving,” arXiv preprint arXiv: 1911.09712, 2019.

[2] C. Godard, O. Mac Aodha, and G. J. Brostow, “Unsupervised monocular depth estimation with left-right consistency,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 270-279.

[3] K. Yamaguchi, D. McAllester, and R. Urtasun, “Efficient joint segmentation, occlusion labeling, stereo and flow estimation,” in European Conference on Computer Vision. Springer, Cham, 2014, pp. 756-771.

[4] Z. Qin, J. Wang, and Y. Lu, ”Monogrnet: A geometric reasoning network for monocular 3d object localization,” in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, no. 01, pp. 8851-8858.

[5] G. Brazil and X. Liu, “M3d-rpn: Monocular 3d region proposal network for object detection,” in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 9287-9296.

[6] Y. Wang, W. L. Chao, D. Garg, B. Hariharan, M. Campbell, and K. Q. Weinberger, “Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 8445-8453.

[7] X. Ma, Z. Wang, H. Li, P. Zhang, W. Ouyang, and X. Fan, “Accurate monocular 3d object detection via color-embedded 3d reconstruction for autonomous driving,” in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 6851-6860.

[8] A. Geiger, P. Lenz, and R. Urtasun, “Are we ready for autonomous driving? the kitti vision benchmark suite,” in 2012 IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 3354-3361.

[9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” arXiv preprint arXiv:1706.03762, 2017.

[10] S. Shi, C. Guo, L. Jiang, Z. Wang, J. Shi, and X. Wang, “PV-RCNN: Point-Voxel Feature Set Abstraction for 3D Object Detection,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 10529-10538.

[11] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, “Dynamic graph cnn for learning on point clouds,” ACM Transactions on Graphics (TOG), 2019, 38(5): 1-12.

[12] H. Zhao, L. Jiang, J. Jia, P. Torr, and V. Koltun, “Point Transformer,” arXiv preprint arXiv: 2012.09164, 2020.

[13] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao, “Deep ordinal regression network for monocular depth estimation,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 2002-2011.

[14] Cai, Z. , and N. Vasconcelos, “Cascade R-CNN: Delving Into High Quality Object Detection,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 6154-6162.